The embodiments of the invention in which an exclusive property or privilege is claimed are defined as follows:

1. In a system for executing parallel jobs, each parallel job comprising multiple tasks executable in parallel by separate resources said system comprising a resource manager for receiving the jobs and dispatching the parallel tasks of the job to the resources through a job launcher unit, a device to facilitate execution of the tasks of a parallel job comprising:

a task starter associated with each task, each task starter commencing execution, on an associated resource, of the task collecting a process identifier of the task from the associated resource, and sending the process identifier of the task to the resource manager;

an external event unit, associated with the resource manager, for receiving the process identifier of the tasks from the task starters.

2. The device as defined in claim 1 further comprising:

a coordinating unit, associated with the resource manager, for inserting said task starters in a command instruction associated with the task, each command instruction corresponding to a task and at least a portion of the command instruction being stored in the resource manager; and

wherein the task starters return the portion of the command instruction of the task with the process identifier.

3. The device as defined in claim 2 wherein the coordinating unit inserts in the command instruction location information for the task starters to send the process identifier and an exit status of the tasks to the resource manager.

4. The device as defined in claim 3 wherein the location information comprises port and host information of the resource manager.

5. The device as defined in claim 2 wherein the command instructions comprise a host identifier for identifying a host containing the resource to execute the task, and, wherein the portion of the command instruction of the task includes the host identifier.

6. The device as defined in claim 5 wherein the coordinating unit of the resource manager starts the job launcher unit which executes the command instructions and starts the task starters on hosts identified by the host identifier.

7. The device as defined in claim 6 wherein once a task has been completed on the associated resource, the task starter collects the exit status of the task from the associated resource and sends the exit status of the task to the resource manager together with the process identifier and host identifier.

8. The device as defined in claim 2 wherein the external event unit performs a commencement action upon receipt of the process identifier from a task starter, said commencement action including mapping the process identifier with the portion of the command instruction stored in the resource manager, and storing the process identifier in the resource manager.

9. The device as defined in claim 8 wherein once a task has been completed on the associated resource, the task starter

collects the exit status of the task from the associated resource and sends the exit status of the task to the resource manager together with the process identifier; and

wherein the external event unit performs a task exit action upon receipt of the task exit action having been predetermined by the coordinating unit.

10. The device as defined in claim 9 wherein the coordinating unit stores the commencement actions and completion actions in the resource manager prior to the task being dispatched, said actions being associated with said command instruction for the task.

11. In a system for executing parallel jobs, each parallel job comprising multiple tasks executable in parallel by separate resources to produce an exit status for the task, said system comprising a resource manager for receiving the jobs and dispatching the tasks to the resources, a method for facilitating execution of the multiple tasks of a parallel job comprising:

for each task, dispatching the task together with an associated task starter from the resource manager to a selected resource for execution of the task;

for each task starter, collecting a process identifier of the task being executed from the resources; and

for each task starter, sending the process identifier to an external event unit associated with the resource manager.

12. The method as defined in claim 11 further comprising: prior to dispatching the task, generating a command instruction corresponding to each task said command instruction including the task starter, and

storing at least a portion of the command instruction in the resource manager;

sending the portion of the command instruction with the process identifier from each task starter to the external event unit;

correlating the command instructions stored in the resource manager with the process identifier and the portion of the command instruction.

13. The method as defined in claim 12 wherein the command instructions include instructions for the task starters to send the process identifiers of the tasks to a location associated with the resource manager; and

wherein the task starters send the process identifiers of the tasks to the location identified by the command instructions upon commencement of the execution of the task.

14. The method as defined in claim 12 wherein the command instructions include instructions for the task starters to send the exit status of the tasks to a location associated with the resource manager; and

wherein each task starter sends the exit status of the task with the process identifier or the portion of the command instruction, or both, to the location identified by the command instructions upon completion of the execution of the task.

15. The method as defined in claim 14 further comprising:

prior to dispatching the task, storing an action to be performed upon receipt by the external event unit of the exit status of a task, said action being associated with the command instruction; and

upon receipt of the exit status of the task, identifying

the action associated with the command instruction of the exit status of the task, and executing the action on the exit status of the task.

16. The method as defined in claim 15 wherein the task starters collect from the resources information regarding resource usage of the task and the task starter sends the information regarding resource usage of the task with the process identifier to the resource manager.

17. The method as defined in claim 12 wherein the command instructions comprise a host identifier for identifying the host containing the resource to execute the task; and

wherein said step of dispatching the task further comprises:

starting a job launcher unit, said job launcher unit executing the command instruction for the task and starting the task starters on resources hosted by the host identified by the command instruction.

18. A system for executing parallel jobs, each parallel job comprising multiple tasks being executed in parallel by separate resources to produce an exit status for the task, said system comprising:

a resource manager for receiving the jobs and selecting resources to execute the multiple tasks of the job;

a task starter associated with each task, each task starter commencing, on an associated resource, the tasks sent from the resource manager, collecting a process identifier from the associated resource, and sending the process identifier of the task to the resource manager.

19. The system as defined in claim 18 further comprising:

a job launcher unit for receiving the multiple tasks of a job from the resource manager and starting the task starters on hosts containing the resources selected by the resource manager.

20. The system as defined in claim 19 further comprising:

a coordinating unit, associated with the resource manager, for inserting in a command instruction for each task the task starters associated with the task and each command instruction corresponding to a task.

21. The system as defined in claim 20 further comprising:

an external event unit, associated with the resource manager for receiving the process identifier and exit status of the task from the task starter;

wherein the external event unit performs a commencement action upon receipt of the process identifier from a task starter, and, a task exit action upon receipt of an exit status of the task; and

wherein the task commencement action and task exit action are predetermined by the coordinating unit.

22. The system as defined in claim 18 wherein the task starters send resource usage information to the resource managers at completion of the task.

23. The device as defined in claim 9 wherein, once a task has been completed on an associated resource, the task starter collects resource usage information of the task from the resource and sends the resource usage information to the resource manager together with the process identifier.